CSc 174 Database Management Systems

5. SQL (Review)

Ying Jin
Computer Science Department
California state University, Sacramento

CREATE TABLE

```
CREATE TABLE   DEPT
(    DNAME          VARCHAR(10) NOT NULL,
     DNUMBER        INTEGER       NOT NULL CHECK
   (DNUMBER>0 AND DNUMBER <21),
     MGRSSN         CHAR(9),
     MGRSTARTDATE       CHAR(9),
     PRIMARY KEY (DNUMBER),
     UNIQUE (DNAME),
     FOREIGN KEY (MGRSSN) REFERENCES EMP  );
```

DROP TABLE

- Remove a relation (base table) and its definition
- The relation can no longer be used in queries, updates, or any other commands
- Example:
  DROP TABLE DEPENDENT;

ALTER TABLE

- Example:

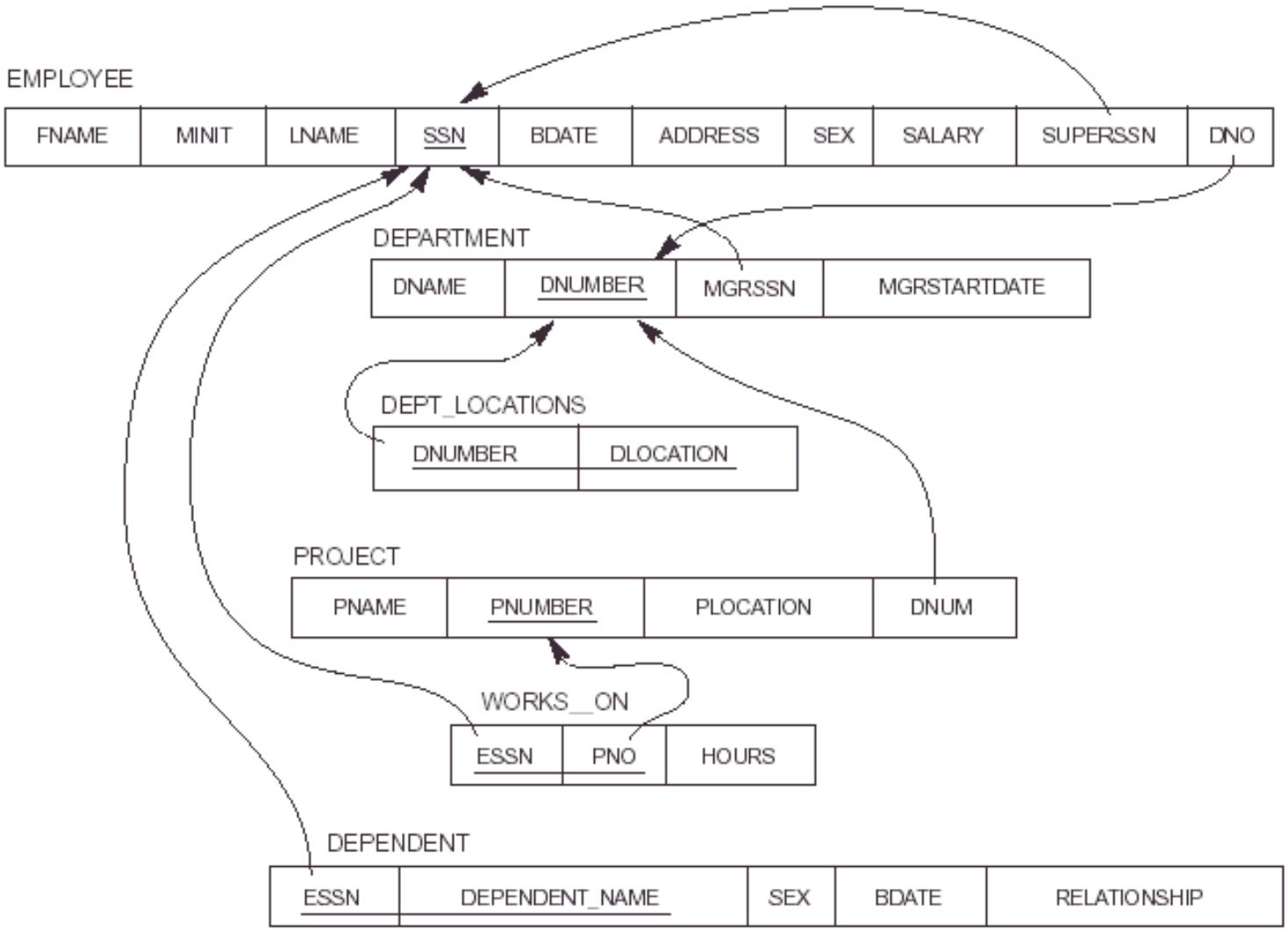
  ALTER TABLE EMPLOYEE ADD JOB VARCHAR(12);

- How to enter value for the new attribute ?

Retrieval Queries in SQL (cont.)

SELECT <attribute list>
FROM <table list>
WHERE <condition>

Relational Database Schema



Simple SQL Queries

- Query 1: Retrieve the name and address of all employees who work for the 'Research' department.
  SQL query?

Simple SQL Queries (cont.)

- Query 2: For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, and birthdate.

Q2:

```
SELECT PNUMBER, DNUM, LNAME, BDATE
FROM   ?
WHERE?
```

Qualify attribute name

- Use the same name for two (or more) attributes as long as the attributes are in *different relations*
- *Qualify* the attribute name with the relation name by *prefixing* the relation name to the attribute name

Example:

- EMPLOYEE.LNAME, DEPARTMENT.DNAME

ALIASES

- Query 8: For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.

```
Q8: SELECT   E.FNAME, E.LNAME, S.FNAME,
                  S.LNAME
    FROM     EMPLOYEE E S
    WHERE    E.SUPERSSN=S.SSN
```

- We can think of E and S as two different copies of EMPLOYEE
  - E represents employees in role of *supervisees*
  - S represents employees in role of *supervisors*

ALIASES (cont.)

- Aliasing can also be used in any SQL query for convenience
  Can also use the AS keyword to specify aliases

```
Q8:   SELECT   E.FNAME, E.LNAME, S.FNAME,
                S.LNAME
      FROM     EMPLOYEE AS E, EMPLOYEE AS S
      WHERE    E.SUPERSSN=S.SSN
```

USE OF DISTINCT

- **Q11:** **SELECT SALARY**
  FROM EMPLOYEE

- **Q11A:** **SELECT DISTINCT SALARY**
  FROM EMPLOYEE

- Which query will return a set ?

NESTING OF QUERIES

- Query 1: Retrieve the name and address of all employees who work for the 'Research' department.

```
Q1: SELECT    FNAME, LNAME, ADDRESS
    FROM      EMPLOYEE
    WHERE     DNO IN
              (SELECT    DNUMBER
              FROM       DEPARTMENT
              WHERE      DNAME='Research' );
```

THE EXISTS FUNCTION

- Check whether the result of a correlated nested query is empty

THE EXISTS FUNCTION (cont.)

- Query 12: Retrieve the name of each employee who has a dependent with the same first name as the employee.

```
Q12B:   SELECT   FNAME, LNAME
        FROM     EMPLOYEE
        WHERE    EXISTS
                 (SELECT   *
                 FROM      DEPENDENT
                 WHERE     SSN=ESSN AND
                 FNAME=DEPENDENT_NAME)
```

AGGREGATE FUNCTIONS

- Include **COUNT**, **SUM**, **MAX**, **MIN**, and **AVG**
- Query 16: Find the maximum salary, the minimum salary, and the average salary among employees who work for the 'Research' department.

THE GROUP BY, HAVING-CLAUSE

- Retrieve the values of these functions for only those groups *that satisfy certain conditions*
- The HAVING-clause
  - Specify a selection condition on groups (rather than on individual tuples)

THE GROUP BY, HAVING-CLAUSE (Cont.)

- Query 22: For each department which has *more than two employees*, retrieve the department number, the number of employees in the department, and their average salary.

ARITHMETIC OPERATIONS

- +, -, *, /
- Query 27: Show the effect of giving all employees who work on the 'ProductX' project a 10% raise.

- Are the salary different after execute the query?

ORDER BY

- The **ORDER BY** clause is used to sort the tuples in a query result based on the values of some attribute(s)
- Query 28: Retrieve a list of employees and the projects each works in, ordered by the employee's department, and within each department ordered alphabetically by employee last name.

```
Q28:
SELECT      DNAME, LNAME, FNAME, PNAME
FROM        DEPARTMENT, EMPLOYEE,
            WORKS_ON, PROJECT
WHERE       DNUMBER=DNO AND SSN=ESSN AND
            PNO=PNUMBER
ORDER BY    DNAME, LNAME
```

Summary of SQL Queries

- A query in SQL can consist of up to six clauses, but only the first two, SELECT and FROM, are mandatory. The clauses are specified in the following order:

SELECT <attribute list>
FROM <table list>
[WHERE <condition>]
[GROUP BY <grouping attribute(s)>]
[HAVING <group condition>]
[ORDER BY <attribute list>]

Specifying Updates in SQL

- There are three SQL commands to modify the database; INSERT, DELETE, and UPDATE

INSERT

- Example1:

```
INSERT INTO  EMPLOYEE
VALUES ('Richard','K','Marini', '653298653',
        '30-DEC-52', '98 Oak Forest,Katy,TX',   'M',
        37000,'987654321', 4 )
```

DELETE

- **U4A:** **DELETE FROM EMPLOYEE**
  WHERE LNAME='Brown'

  U4D: **DELETE FROM EMPLOYEE**

UPDATE

- Example: Change the location and controlling department number of project number 10 to 'Bellaire' and 5, respectively.

```
U5:  UPDATE      PROJECT
     SET         PLOCATION = 'Bellaire', DNUM = 5
     WHERE       PNUMBER=10
```

UPDATE (cont.)

- Example: Give all employees in the 'Research' department a 10% raise in salary.

```
U6: UPDATE    EMPLOYEE
    SET       SALARY = SALARY *1.1
    WHERE     DNO  IN (SELECT DNUMBER
                  FROM    DEPARTMENT
                  WHERE   DNAME='Research')
```

These slides are based on the textbook:

R. Elmaseri and S. Navathe, *Fundamentals of Database System*, 7th Edition, Addison-Wesley.